

ΣΧΕΔΙΑΣΜΌΣ ΣΥΣΤΗΜΆΤΩΝ VLSI

5^η Εργαστηριακή Άσκηση

Γιώργος Ντάκος 1059569

7 Ιουνίου 2021

Περίληψη

Στο παρακάτω κείμενο παραδίδεται η αναφορά της 5^{ης} εργαστηριακής άσκησης του μαθήματος Σχεδιασμός Συστημάτων VLSI . Όλοι οι κώδικες έχουν συγγραφεί σε γλώσσα **VHDL & C++** μέσω του **Notepad++ & HLS** και έχουν μεταγλωττιστεί και επιβεβαιωθεί οι λειτουργίες τους μέσω του εργαλείου **ModelSim HLS**. Τέλος η σύνθεση όλων των κυκλωμάτων πραγματοποιήθηκαν από το εργαλείο **VIVADO & HLS** της **Xilinx**.

Περιεχόμενα

Ενότητα Α	4
Ενότητα Β	16
Ενότητα Β	16

Κατάλογος Σχημάτων

1	Αποτελέσματα simulation	4
2	Αποτελέσματα csimulation	5
3	Αποτελέσματα σύνθεσης	6
4	Αποτελέσματα σύνθεσης	6
5	Αποτελέσματα σύνθεσης	7
6	Αποτελέσματα cosimulation	7
7	Αποτελέσματα rtl	8
8	Αποτελέσματα σύνθεσης	10
9	Αποτελέσματα σύνθεσης	11
10	Αποτελέσματα σύνθεσης	12
11	Αποτελέσματα σύνθεσης	13
12	Αποτελέσματα σύνθεσης	14
13	Αποτελέσματα cosimulation	14
14	Αποτελέσματα rtl	15
15	Αποτελέσματα csimulation	16
16	Αποτελέσματα csimulation	17
17	Αποτελέσματα συνθεσης	18
18	Αποτελέσματα συνθεσης	19
19	Αποτελέσματα συνθεσης	20
20	Αποτελέσματα cosimulation	21
21	Αποτελέσματα rtl	22
22	Αποτελέσματα rtl	24
23	Αποτελέσματα rtl	25
24	Αποτελέσματα rtl	26
25	Αποτελέσματα rtl	27
26	Αποτελέσματα rtl	28
27	Αποτελέσματα rtl	29
28	Αποτελέσματα rtl	30

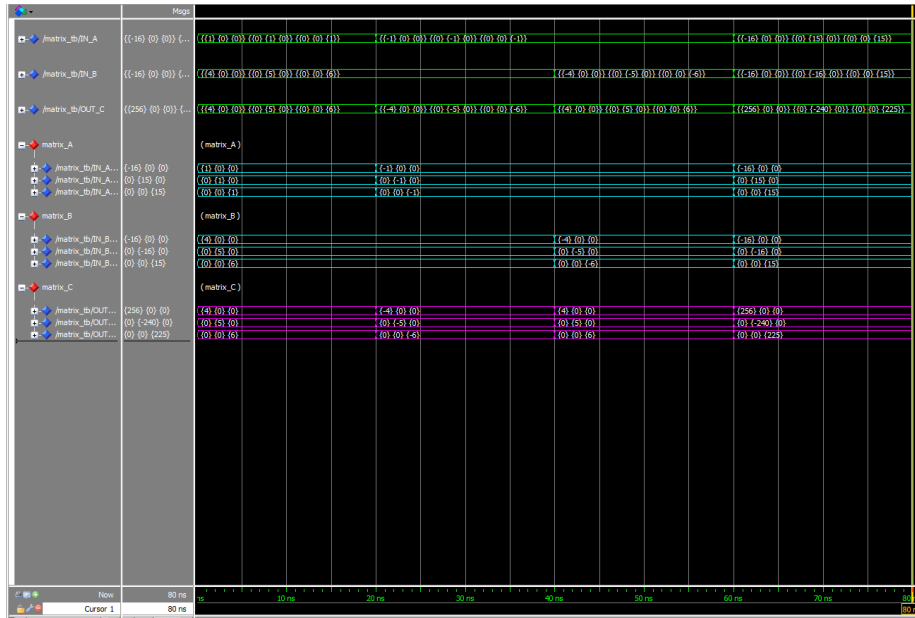
Κατάλογος Πινάκων

1	PowerSupply	4
2	Utiliazation	4
3	Data for Timing	5

Ενότητα A

A.1

Στην συγκεκριμένη ενότητα θα φτιάξουμε έναν πολλαπλασιασμό wise element 2 μητρώων 3x3 των 5 bits το κάθε στοιχείο. Άρα στο αποτέλεσμα θα έχουμε ένα άλλο μητρώο 3x3 των 10(5+5) bits το κάθε στοιχείο. Αφού δημιουργήσουμε το αρχείο του κώδικα πραγματοποιούμε την εξομοίωση του και παίρνουμε τα παρακάτω αποτελέσματα όπου και μας επιβεβαιώνουν για την σωστή λειτουργία του κώδικα



Σχήμα 1: Αποτελεσματα simulation

Μετά κάνουμε την σύνθεση του κυκλωματός και παίρνουμε ως αποτέλεσμα για τους πόρους κτλ. που χρησιμοποίησε το σύστημα μας.(σχηματικό της σύνθεσης στο τέλος της εργασίας)

POWER												
Name of circuit	Dynamic	Signal	Logic	I/O	Block	Junction Temperature	Power supplied to all-chip devices	Thermal Margin	Effective JA	Confidence level	Total On-Chip Power	
Bits5MatrixMultiplier3X3	92.900 W(100%)	2.304 W(2%)	1.000 W(2%)	60.92 W(65%)	1.974 W(2%)	100°C (High Temperature)	0 W	-68.8 J(330 W) High P	1.0 C/W	1.000	14.004 W(Temp. assumed)	

Table 1: PowerSupply

Utilization			
Name of circuit	LUT	IO	FF
Bits5MatrixMultiplier3X3	225 of 78600 (0.29%)	180 of 250 (72%)	0

Table 2: Utilization

Timing	Max Delay	
Name of Circuit	Critical Path	Frequency
Bits5MatrixMultiplier3X3	12.209 ns	81.90 MHz

Table 3: Data for Timing

A.2

Τώρα θα φτιάξουμε την ίδια διαδικασία σε γλώσσα c κάτι το οποίο είναι πιο εύκολο και πρακτικό. Δημιουργώντας την κατάλληλη συνάρτηση και το αρχείο testbench τρέχουμε το csimulation και παίρνουμε παρακάτω αποτελέσματα τα οποία μας δείχνουν την ορθή λειτουργία του κωδικά μας.

```

1 INFO: [SIM 2] ***** CSIM start *****
2 INFO: [SIM 4] CSIM will launch GCC as the compiler.
3 make: 'csim.exe' is up to date.
4 A[0][0] = 6
5 B[0][0] = 7
6 C[0][0] = 42
7 C_test[0][0] = 42
8 A[0][1] = -10
9 B[0][1] = 5
10 C[0][1] = -50
11 C_test[0][1] = -50
12 A[0][2] = -9
13 B[0][2] = -11
14 C[0][2] = 99
15 C_test[0][2] = 99
16 A[1][0] = 13
17 B[1][0] = -3
18 C[1][0] = -39
19 C_test[1][0] = -39
20 A[1][1] = -14
21 B[1][1] = -12
22 C[1][1] = 168
23 C_test[1][1] = 168
24 A[1][2] = -3
25 B[1][2] = 4
26 C[1][2] = -12
27 C_test[1][2] = -12
28 A[2][0] = -10
29 B[2][0] = -7
30 C[2][0] = 70
31 C_test[2][0] = 70
32 A[2][1] = -7
33 B[2][1] = -15
34 C[2][1] = 105
35 C_test[2][1] = 105
36 A[2][2] = -15
37 B[2][2] = 13
38 C[2][2] = -195
39 C_test[2][2] = -195
40 INFO: [SIM 1] CSim done with 0 errors.
41 TIME: [SIM 3] ***** CSIM finish *****

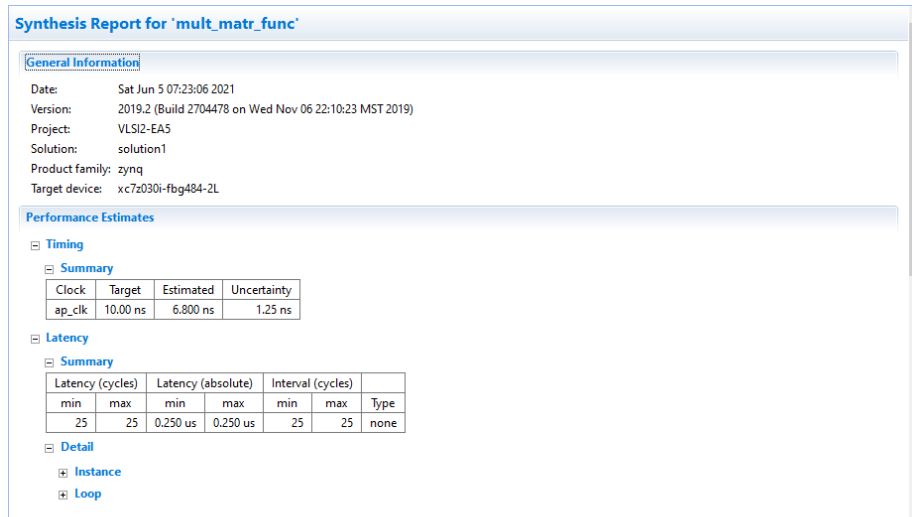
```

Σχήμα 2: Αποτελέσματα csimulation

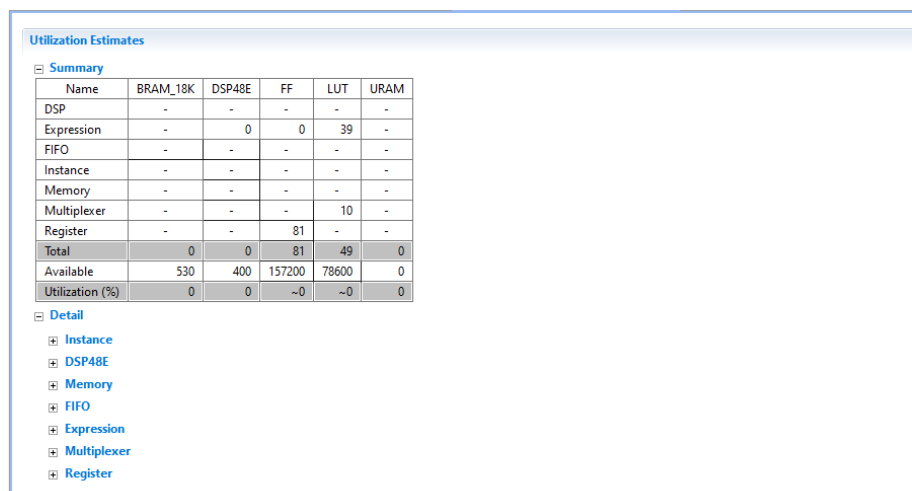
Στην συνέχεια ακολουθούν τα αποτελέσματα της σύνθεσης για την συναρτησή μας.

Έπειτα εκτέλεσαμε το cosimulation και πήραμε τα παρακάτω αποτελέσματα:

Επιπλέον μας δημιούργησε ένα αρχείο κώδικα σε VHDL που έχει αρκετές διαφορές από τον δικό μας κώδικα. Με μια γρήγορη ματιά παρατηρούμε ότι ο κώδικας που πάραξε το εργαλείο HLS είναι πολύ μεγαλύτερος από τον δικό μας. Έπειτα παρατηρούμε ότι μια πληθώρα από σήματα εξόδους και εισόδους στο κομμάτι της οντότητας ενώ εμείς χρειαστήκαμε μόνο 2 εισόδους και 1 έξοδο. Στη συνέχεια βλέπουμε ότι στην αρχιτεκτονική του συστήματος αυτού έχουμε πολλά τοπικά σήματα της μονάδας και μια διαφορετική διαδικασία με την οποία περιγράφει την



Σχήμα 3: Αποτελέσματα σύνθεσης



Σχήμα 4: Αποτελέσματα σύνθεσης

Interface

Summary

RTL Ports	Dir	Bits	Protocol	Source Object	C Type
ap_clk	in	1	ap_ctrl_hs	mult_matr_func	return value
ap_rst	in	1	ap_ctrl_hs	mult_matr_func	return value
ap_start	in	1	ap_ctrl_hs	mult_matr_func	return value
ap_done	out	1	ap_ctrl_hs	mult_matr_func	return value
ap_idle	out	1	ap_ctrl_hs	mult_matr_func	return value
ap_ready	out	1	ap_ctrl_hs	mult_matr_func	return value
A_V_address0	out	4	ap_memory	A_V	array
A_V_ce0	out	1	ap_memory	A_V	array
A_V_q0	in	5	ap_memory	A_V	array
B_V_address0	out	4	ap_memory	B_V	array
B_V_ce0	out	1	ap_memory	B_V	array
B_V_q0	in	5	ap_memory	B_V	array
C_V_address0	out	4	ap_memory	C_V	array
C_V_ce0	out	1	ap_memory	C_V	array
C_V_we0	out	1	ap_memory	C_V	array
C_V_d0	out	10	ap_memory	C_V	array

Export the report(.html) using the [Export Wizard](#)

Open Analysis Perspective [Analysis Perspective](#)

Σχήμα 5: Αποτελέσματα σύνθεσης

Cosimulation Report for 'mult_matr_func'

Result

RTL	Status	Latency			Interval		
		min	avg	max	min	avg	max
VHDL	Pass	25	25	25	NA	NA	NA
Verilog	NA	NA	NA	NA	NA	NA	NA

Export the report(.html) using the [Export Wizard](#)

Σχήμα 6: Αποτελέσματα cosimulation

λειτουργία του κυκλώματος μας αλλά στην ουσία και οι 2 κώδικες πραγματοποιούν την ίδια λειτουργία.

Τέλος πραγματοποιούμε το export RTL και παίρνουμε τους τελικούς πόρους του συστήματός μας.

Export Report for 'mult_matr_func'

General Information

Report date: Sat Jun 05 07:29:02 +0300 2021
Project: VLSI2-EA5
Solution: solution1
Device target: xc7z030i-fbg484-2L
Implementation tool: Xilinx Vivado v.2019.2

Resource Usage

	VHDL
SLICE	0
LUT	41
FF	20
DSP	0
BRAM	0
SRL	0

Final Timing

	VHDL
CP required	10.000
CP achieved post-synthesis	1.367

Timing met

Export the report(.html) using the [Export Wizard](#)

Σχήμα 7: Αποτελέσματα rtl

A.3

Για να ελαττώσουμε το latency όσο πιο πολύ μπορούμε θα χρησιμοποιήσουμε τις ετικέτες pragma στον κωδικά μας. Αυτό που πρέπει να αναφέρουμε είναι ότι δεν λάβαμε υπόψη μας περιορισμούς στους πόρους του συστήματός μας κάτι που στην πραγματικότητα είναι αρκετά σημαντικό κομμάτι στην υλοποίηση κυκλωμάτων. Αυτό που κάναμε για αρχή είναι να σπάσουμε τους 3 πίνακες σε 3 διαδοχικές διαφορετικές γραμμές των 3 στοιχείων, οι οποίες θα μπορούν να επεξεργαστούν παράλληλα. Έπειτα βλέπουμε ότι έχουμε εμφολευμένο for σε ένα μπλοκ for οπότε η χρήση pipelined μας βολεύει αρκετά για να μειώσουμε αρκετά το latency του συστήματός μας, όπου η κάθε διαδοχική επανάληψη του βρόχου θα γίνεται μετά από 1 κύκλο ρολογιού και το εμφολευμένο for θα γίνει total unroll οπότε όλες οι επαναλήψεις του θα τρέξουν παράλληλα. Τα αποτελέσματα των synthesis, cosimulation, export rtl φαίνονται παρακάτω:

Απο τα αποτελέσματα παρατηρούμε ότι μειώσαμε το latency 3 φορές και λίγο παραπάνω απο το σύστημα που δεν βάλαμε pragmas. Βέβαια παρατηρούμε ότι χρειάστηκαν τα διπλάσια και παραπάνω LUT αλλά ταυτόχρονα σχεδόν υποδιπλασιάστηκαν τα FF που θα χρησιμοποιήσουμε. Τώρα ο κώδικας VHDL που δημιούργησε είναι εξίσου διαφορετικός από τους άλλους δύο. Πρώτα πόλλα δημιουργήθηκαν 2 αρχεία VHDL όπου το σύστημα του το καλούμε στο τελικό μας σύστημα. Επίσης δηλώθηκαν πολλές περισσότεροι είσοδοι και έξοδοι κάτι το οποίο περιμέναμε αφού σπάσαμε τους πίνακες μας σε διαφορετικά τμήματα. Τέλος η αρχιτεκτονική έχει αλλάξει κατά πολύ, με περισσότερα τοπικά σήματα και την χρήση του υποσυστήματος για λειτουργία του τελικού συστήματος μας. Βέβαια η λειτουργία όλων των κωδικών VHDL είναι η ίδια.

Synthesis Report for 'mult_matr_func'

General Information

Date: Sat Jun 5 18:34:58 2021
Version: 2019.2 (Build 2704478 on Wed Nov 06 22:10:23 MST 2019)
Project: VLSI2-EA5
Solution: solution1
Product family: zynq
Target device: xc7z030i-fbg484-2L

Performance Estimates

Timing

Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00 ns	8.160 ns	1.25 ns

Latency

Summary

Latency (cycles)		Latency (absolute)		Interval (cycles)		
min	max	min	max	min	max	Type
8	8	80.000 ns	80.000 ns	8	8	none

Detail

+ Instance

+ Loop

Σχήμα 8: Αποτελέσματα σύνθεσης

Utilization Estimates

[-] Summary

Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	-	-	-	-
Expression	-	0	0	57	-
FIFO	-	-	-	-	-
Instance	-	-	0	52	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	49	-
Register	-	-	11	-	-
Total	0	0	11	158	0
Available	530	400	157200	78600	0
Utilization (%)	0	0	~0	~0	0

[-] Detail

- + Instance
- + DSP48E
- + Memory
- + FIFO
- + Expression
- + Multiplexer
- + Register

Σχήμα 9: Αποτελέσματα σύνθεσης

Interface

Summary

RTL Ports	Dir	Bits	Protocol	Source Object	C Type
ap_clk	in	1	ap_ctrl_hs	mult_matr_func	return value
ap_rst	in	1	ap_ctrl_hs	mult_matr_func	return value
ap_start	in	1	ap_ctrl_hs	mult_matr_func	return value
ap_done	out	1	ap_ctrl_hs	mult_matr_func	return value
ap_idle	out	1	ap_ctrl_hs	mult_matr_func	return value
ap_ready	out	1	ap_ctrl_hs	mult_matr_func	return value
A_0_V_address0	out	2	ap_memory	A_0_V	array
A_0_V_ce0	out	1	ap_memory	A_0_V	array
A_0_V_q0	in	5	ap_memory	A_0_V	array
A_0_V_address1	out	2	ap_memory	A_0_V	array
A_0_V_ce1	out	1	ap_memory	A_0_V	array
A_0_V_q1	in	5	ap_memory	A_0_V	array
A_1_V_address0	out	2	ap_memory	A_1_V	array
A_1_V_ce0	out	1	ap_memory	A_1_V	array
A_1_V_q0	in	5	ap_memory	A_1_V	array
A_1_V_address1	out	2	ap_memory	A_1_V	array
A_1_V_ce1	out	1	ap_memory	A_1_V	array
A_1_V_q1	in	5	ap_memory	A_1_V	array
A_2_V_address0	out	2	ap_memory	A_2_V	array
A_2_V_ce0	out	1	ap_memory	A_2_V	array
A_2_V_q0	in	5	ap_memory	A_2_V	array
A_2_V_address1	out	2	ap_memory	A_2_V	array
A_2_V_ce1	out	1	ap_memory	A_2_V	array
A_2_V_q1	in	5	ap_memory	A_2_V	array
B_0_V_address0	out	2	ap_memory	B_0_V	array
B_0_V_ce0	out	1	ap_memory	B_0_V	array
B_0_V_q0	in	5	ap_memory	B_0_V	array
B_0_V_address1	out	2	ap_memory	B_0_V	array

Σχήμα 10: Αποτελέσματα σύνθεσης

B_0_V_ce1	out	1	ap_memory	B_0_V	array
B_0_V_q1	in	5	ap_memory	B_0_V	array
B_1_V_address0	out	2	ap_memory	B_1_V	array
B_1_V_ce0	out	1	ap_memory	B_1_V	array
B_1_V_q0	in	5	ap_memory	B_1_V	array
B_1_V_address1	out	2	ap_memory	B_1_V	array
B_1_V_ce1	out	1	ap_memory	B_1_V	array
B_1_V_q1	in	5	ap_memory	B_1_V	array
B_2_V_address0	out	2	ap_memory	B_2_V	array
B_2_V_ce0	out	1	ap_memory	B_2_V	array
B_2_V_q0	in	5	ap_memory	B_2_V	array
B_2_V_address1	out	2	ap_memory	B_2_V	array
B_2_V_ce1	out	1	ap_memory	B_2_V	array
B_2_V_q1	in	5	ap_memory	B_2_V	array
C_0_V_address0	out	2	ap_memory	C_0_V	array
C_0_V_ce0	out	1	ap_memory	C_0_V	array
C_0_V_we0	out	1	ap_memory	C_0_V	array
C_0_V_d0	out	10	ap_memory	C_0_V	array
C_0_V_address1	out	2	ap_memory	C_0_V	array
C_0_V_ce1	out	1	ap_memory	C_0_V	array
C_0_V_we1	out	1	ap_memory	C_0_V	array
C_0_V_d1	out	10	ap_memory	C_0_V	array
C_1_V_address0	out	2	ap_memory	C_1_V	array
C_1_V_ce0	out	1	ap_memory	C_1_V	array
C_1_V_we0	out	1	ap_memory	C_1_V	array
C_1_V_d0	out	10	ap_memory	C_1_V	array
C_1_V_address1	out	2	ap_memory	C_1_V	array
C_1_V_ce1	out	1	ap_memory	C_1_V	array
C_1_V_we1	out	1	ap_memory	C_1_V	array

Σχήμα 11: Αποτελέσματα σύνθεσης

C_1_V_d1	out	10	ap_memory	C_1_V	array
C_2_V_address0	out	2	ap_memory	C_2_V	array
C_2_V_ce0	out	1	ap_memory	C_2_V	array
C_2_V_we0	out	1	ap_memory	C_2_V	array
C_2_V_d0	out	10	ap_memory	C_2_V	array
C_2_V_address1	out	2	ap_memory	C_2_V	array
C_2_V_ce1	out	1	ap_memory	C_2_V	array
C_2_V_we1	out	1	ap_memory	C_2_V	array
C_2_V_d1	out	10	ap_memory	C_2_V	array

Σχήμα 12: Αποτελέσματα σύνθεσης

Cosimulation Report for 'mult_matr_func'							
Result							
		Latency			Interval		
RTL	Status	min	avg	max	min	avg	max
VHDL	Pass	8	8	8	NA	NA	NA
Verilog	NA	NA	NA	NA	NA	NA	NA

Export the report(.html) using the [Export Wizard](#)

Σχήμα 13: Αποτελέσματα cosimulation

Export Report for 'mult_matr_func'

General Information

Report date: Sat Jun 05 18:42:40 +0300 2021
Project: VLSI2-EA5
Solution: solution1
Device target: xc7z030i-fbg484-2L
Implementation tool: Xilinx Vivado v.2019.2

Resource Usage

	VHDL
SLICE	0
LUT	102
FF	11
DSP	0
BRAM	0
SRL	0

Final Timing

	VHDL
CP required	10.000
CP achieved post-synthesis	1.505

Timing met

Σχήμα 14: Αποτελέσματα rtl

Ενότητα B

B.1

Στην συγκεκριμένη άσκηση πρέπει να φτιάξουμε τον πολλαπλασιασμό 2 μητρώων 4x4 με 12 bits εύρος το κάθε στοιχείο οπότε στα αποτελέσματα θα έχουμε ένα μητρώο 4x4 με το κάθε στοιχείο να έχει εύρος 24 bits (12+12). Πραγματοποιούμε το simulation και παίρνουμε τα εξής αποτελέσματα:

Ενότητα B

B.1

```
1 INFO: [SIM 2] ***** CSIM start *****
2 INFO: [SIM 4] CSIM will launch GCC as the compiler.
3   Compiling ../../matrix_mult_fun_tb.cpp in debug mode
4   Generating csim.exe
5 A[0][0] = 38
6 B[0][0] = -473
7 C[0][0] = 2814492
8 C_test[0][0] = 2814492
9 A[0][1] = 758
10 B[0][1] = -1659
11 C[0][1] = -1947458
12 C_test[0][1] = -1947458
13 A[0][2] = 663
14 B[0][2] = -491
15 C[0][2] = -471881
16 C_test[0][2] = -471881
17 A[0][3] = 173
18 B[0][3] = -483
19 C[0][3] = 614552
20 C_test[0][3] = 614552
21 A[1][0] = -1838
22 B[1][0] = 1940
23 C[1][0] = 6305043
24 C_test[1][0] = 6305043
25 A[1][1] = 1757
26 B[1][1] = -572
27 C[1][1] = -7522
28 C_test[1][1] = -7522
29 A[1][2] = 1142
30 B[1][2] = 281
31 C[1][2] = 366711
32 C_test[1][2] = 366711
33 A[1][3] = 57
34 B[1][3] = -463
35 C[1][3] = 1626753
36 C_test[1][3] = 1626753
37 A[2][0] = 753
38 B[2][0] = 1709
39 C[2][0] = -86040
40 C_test[2][0] = -86040
```

Σχήμα 15: Αποτελέσματα csimulation


```

41 A[2][1] = -1099
42 B[2][1] = -1704
43 C[2][1] = -3646583
44 C_test[2][1] = -3646583
45 A[2][2] = 496
46 B[2][2] = -877
47 C[2][2] = -1689284
48 C_test[2][2] = -1689284
49 A[2][3] = 1175
50 B[2][3] = 1330
51 C[2][3] = 1498068
52 C_test[2][3] = 1498068
53 A[3][0] = -2023
54 B[3][0] = 1323
55 C[3][0] = 1799286
56 C_test[3][0] = 1799286
57 A[3][1] = 209
58 B[3][1] = -1856
59 C[3][1] = 3686777
60 C_test[3][1] = 3686777
61 A[3][2] = 1533
62 B[3][2] = -490
63 C[3][2] = 516081
64 C_test[3][2] = 516081
65 A[3][3] = -1650
66 B[3][3] = 590
67 C[3][3] = 1945732
68 C_test[3][3] = 1945732
69 INFO: [SIM 1] CSim done with 0 errors.
70 INFO: [SIM 3] ***** CSIM finish *****
71

```

Σχήμα 16: Αποτελέσματα csimulation

Μετά τα παραπάνω αποτελέσματα και την ορθή λειτουργία της συναρτησής μας πραγματοποιούμε την σύνθεση,cosimulation & rtl export και επιστράφηκαν τα εξής αποτελέσματα :

Synthesis Report for 'OGmult_matr_func'

General Information

Date: Sat Jun 5 21:10:04 2021
Version: 2019.2 (Build 2704478 on Wed Nov 06 22:10:23 MST 2019)
Project: VLSI2-EA5
Solution: solution2
Product family: zynq
Target device: xc7z030i-fbg484-2L

Performance Estimates

Timing

Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00 ns	7.305 ns	1.25 ns

Latency

Summary

Latency (cycles)		Latency (absolute)		Interval (cycles)		
min	max	min	max	min	max	Type
169	169	1.690 us	1.690 us	169	169	none

Detail

Instance

Loop

Σχήμα 17: Αποτελέσματα σύνθεσης

Utilization Estimates

[-] Summary

Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	1	-	-	-
Expression	-	-	0	36	-
FIFO	-	-	-	-	-
Instance	-	-	-	-	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	16	-
Register	-	-	57	-	-
Total	0	1	57	52	0
Available	530	400	157200	78600	0
Utilization (%)	0	~0	~0	~0	0

[-] Detail

- + Instance
- + DSP48E
- + Memory
- + FIFO
- + Expression
- + Multiplexer
- + Register

Σχήμα 18: Αποτελέσματα συνθεσης

Interface

Summary

RTL Ports	Dir	Bits	Protocol	Source Object	C Type
ap_clk	in	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_rst	in	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_start	in	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_done	out	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_idle	out	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_ready	out	1	ap_ctrl_hs	OGmult_matr_func	return value
A_V_address0	out	4	ap_memory	A_V	array
A_V_ce0	out	1	ap_memory	A_V	array
A_V_q0	in	12	ap_memory	A_V	array
B_V_address0	out	4	ap_memory	B_V	array
B_V_ce0	out	1	ap_memory	B_V	array
B_V_q0	in	12	ap_memory	B_V	array
C_V_address0	out	4	ap_memory	C_V	array
C_V_ce0	out	1	ap_memory	C_V	array
C_V_we0	out	1	ap_memory	C_V	array
C_V_d0	out	24	ap_memory	C_V	array

Export the report(.html) using the [Export Wizard](#)

Open Analysis Perspective [Analysis Perspective](#)

Σχήμα 19: Αποτελέσματα σύνθεσης

Cosimulation Report for 'OGmult_matr_func'

Result

RTL	Status	Latency			Interval		
		min	avg	max	min	avg	max
VHDL	Pass	169	169	169	NA	NA	NA
Verilog	NA	NA	NA	NA	NA	NA	NA

Export the report(.html) using the [Export Wizard](#)

Σχήμα 20: Αποτελέσματα cosimulation

Export Report for 'OGmult_matr_func'

General Information

Report date: Sat Jun 05 21:16:01 +0300 2021
Project: VLSI2-EA5
Solution: solution2
Device target: xc7z030i-fbg484-2L
Implementation tool: Xilinx Vivado v.2019.2

Resource Usage

	VHDL
SLICE	0
LUT	27
FF	30
DSP	1
BRAM	0
SRL	0

Final Timing

	VHDL
CP required	10.000
CP achieved post-synthesis	1.660

Timing met

Export the report(.html) using the [Export Wizard](#)

Σχήμα 21: Αποτελέσματα rtl

B.2

Τώρα θα προσπαθήσουμε πάλι να μειώσουμε το latency του αρχικού μας συστήματος με την χρήση pragmas ετικέτων. Πάλι να σημειώσουμε ότι δεν λάβαμε υπόψη μας το πόσους πόρους θα χρησιμοποιήσει το σύστημά μας. Στο συγκεκριμένο κώδικα αυτό που κάναμε είναι να σπάσουμε και τους 3 πίνακες τον κάθε ένα σε μεμονωμένα στοιχεία δηλαδή τα συνολικά 16 στοιχεία του κάθε πίνακα θα γίνουν 16 ξεχωριστοί πίνακες. Στη συνέχεια αφού έχουμε ολά τα στοιχεία του κάθε πίνακα μεμονωμένα θα βόλευε πάρα πολύ να κάνουμε και τους 3 βρόγχους for, totally unroll έτσι όλες οι επαναλήψεις των βρόγχων θα τρέξουν ταυτόχρονα παράλληλα. Θα μπορούσαμε πάλι να βάλουμε pipelined θα πέραμε επίσης αρκετά καλά αποτελέσματα αλλά απο την στιγμή που έχουμε σπάσει ολοκληρωτικά τον κάθε πίνακα το να κάνουμε totally unroll τους 3 βρόγχους θα μας δώσουν τα βέλτιστα αποτελέσματα όσον αφορά το latency . Έχοντας κάνει όλα τα παραπάνω εκτελούμε τα synthesis, cosimulation & rtl export και καταγράφουμε τα παρακάτω αποτελέσματα:

Synthesis Report for 'OGmult_matr_func'

General Information

Date: Sat Jun 5 21:39:54 2021
Version: 2019.2 (Build 2704478 on Wed Nov 06 22:10:23 MST 2019)
Project: VLSI2-EA5
Solution: solution3
Product family: zynq
Target device: xc7z030i-fbg484-2L

Performance Estimates

Timing

Summary

Clock	Target	Estimated	Uncertainty
ap_clk	10.00 ns	8.993 ns	1.25 ns

Latency

Summary

Latency (cycles)		Latency (absolute)		Interval (cycles)		Type
min	max	min	max	min	max	
4	4	40.000 ns	40.000 ns	4	4	none

Detail

Instance

Loop

Σχήμα 22: Αποτελέσματα rtl

Utilization Estimates

Summary

Name	BRAM_18K	DSP48E	FF	LUT	URAM
DSP	-	64	-	-	-
Expression	-	-	0	384	-
FIFO	-	-	-	-	-
Instance	-	-	-	-	-
Memory	-	-	-	-	-
Multiplexer	-	-	-	100	-
Register	-	-	1205	-	-
Total	0	64	1205	484	0
Available	530	400	157200	78600	0
Utilization (%)	0	16	~0	~0	0

Detail

- + Instance
- + DSP48E
- + Memory
- + FIFO
- + Expression
- + Multiplexer
- + Register

Σχήμα 23: Αποτελέσματα rtl

Interface

Summary

RTL Ports	Dir	Bits	Protocol	Source Object	C Type
ap_clk	in	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_rst	in	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_start	in	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_done	out	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_idle	out	1	ap_ctrl_hs	OGmult_matr_func	return value
ap_ready	out	1	ap_ctrl_hs	OGmult_matr_func	return value
A_0_V_address0	out	2	ap_memory	A_0_V	array
A_0_V_ce0	out	1	ap_memory	A_0_V	array
A_0_V_q0	in	12	ap_memory	A_0_V	array
A_0_V_address1	out	2	ap_memory	A_0_V	array
A_0_V_ce1	out	1	ap_memory	A_0_V	array
A_0_V_q1	in	12	ap_memory	A_0_V	array
A_1_V_address0	out	2	ap_memory	A_1_V	array
A_1_V_ce0	out	1	ap_memory	A_1_V	array
A_1_V_q0	in	12	ap_memory	A_1_V	array
A_1_V_address1	out	2	ap_memory	A_1_V	array
A_1_V_ce1	out	1	ap_memory	A_1_V	array
A_1_V_q1	in	12	ap_memory	A_1_V	array
A_2_V_address0	out	2	ap_memory	A_2_V	array
A_2_V_ce0	out	1	ap_memory	A_2_V	array
A_2_V_q0	in	12	ap_memory	A_2_V	array
A_2_V_address1	out	2	ap_memory	A_2_V	array
A_2_V_ce1	out	1	ap_memory	A_2_V	array
A_2_V_q1	in	12	ap_memory	A_2_V	array
A_3_V_address0	out	2	ap_memory	A_3_V	array
A_3_V_ce0	out	1	ap_memory	A_3_V	array
A_3_V_q0	in	12	ap_memory	A_3_V	array

Σχήμα 24: Αποτελέσματα rtl

A_3_V_address1	out	2	ap_memory	A_3_V	array
A_3_V_ce1	out	1	ap_memory	A_3_V	array
A_3_V_q1	in	12	ap_memory	A_3_V	array
B_0_V_address0	out	2	ap_memory	B_0_V	array
B_0_V_ce0	out	1	ap_memory	B_0_V	array
B_0_V_q0	in	12	ap_memory	B_0_V	array
B_0_V_address1	out	2	ap_memory	B_0_V	array
B_0_V_ce1	out	1	ap_memory	B_0_V	array
B_0_V_q1	in	12	ap_memory	B_0_V	array
B_1_V_address0	out	2	ap_memory	B_1_V	array
B_1_V_ce0	out	1	ap_memory	B_1_V	array
B_1_V_q0	in	12	ap_memory	B_1_V	array
B_1_V_address1	out	2	ap_memory	B_1_V	array
B_1_V_ce1	out	1	ap_memory	B_1_V	array
B_1_V_q1	in	12	ap_memory	B_1_V	array
B_2_V_address0	out	2	ap_memory	B_2_V	array
B_2_V_ce0	out	1	ap_memory	B_2_V	array
B_2_V_q0	in	12	ap_memory	B_2_V	array
B_2_V_address1	out	2	ap_memory	B_2_V	array
B_2_V_ce1	out	1	ap_memory	B_2_V	array
B_2_V_q1	in	12	ap_memory	B_2_V	array
B_3_V_address0	out	2	ap_memory	B_3_V	array
B_3_V_ce0	out	1	ap_memory	B_3_V	array
B_3_V_q0	in	12	ap_memory	B_3_V	array
B_3_V_address1	out	2	ap_memory	B_3_V	array
B_3_V_ce1	out	1	ap_memory	B_3_V	array
B_3_V_q1	in	12	ap_memory	B_3_V	array
C_0_V_address0	out	2	ap_memory	C_0_V	array
C_0_V_ce0	out	1	ap_memory	C_0_V	array
C_0_V_we0	out	1	ap_memory	C_0_V	array
C_0_V_d0	out	24	ap_memory	C_0_V	array

Σχήμα 25: Αποτελέσματα rtl

C_0_V_address1	out	2	ap_memory	C_0_V	array
C_0_V_ce1	out	1	ap_memory	C_0_V	array
C_0_V_we1	out	1	ap_memory	C_0_V	array
C_0_V_d1	out	24	ap_memory	C_0_V	array
C_1_V_address0	out	2	ap_memory	C_1_V	array
C_1_V_ce0	out	1	ap_memory	C_1_V	array
C_1_V_we0	out	1	ap_memory	C_1_V	array
C_1_V_d0	out	24	ap_memory	C_1_V	array
C_1_V_address1	out	2	ap_memory	C_1_V	array
C_1_V_ce1	out	1	ap_memory	C_1_V	array
C_1_V_we1	out	1	ap_memory	C_1_V	array
C_1_V_d1	out	24	ap_memory	C_1_V	array
C_2_V_address0	out	2	ap_memory	C_2_V	array
C_2_V_ce0	out	1	ap_memory	C_2_V	array
C_2_V_we0	out	1	ap_memory	C_2_V	array
C_2_V_d0	out	24	ap_memory	C_2_V	array
C_2_V_address1	out	2	ap_memory	C_2_V	array
C_2_V_ce1	out	1	ap_memory	C_2_V	array
C_2_V_we1	out	1	ap_memory	C_2_V	array
C_2_V_d1	out	24	ap_memory	C_2_V	array
C_3_V_address0	out	2	ap_memory	C_3_V	array
C_3_V_ce0	out	1	ap_memory	C_3_V	array
C_3_V_we0	out	1	ap_memory	C_3_V	array
C_3_V_d0	out	24	ap_memory	C_3_V	array
C_3_V_address1	out	2	ap_memory	C_3_V	array
C_3_V_ce1	out	1	ap_memory	C_3_V	array
C_3_V_we1	out	1	ap_memory	C_3_V	array
C_3_V_d1	out	24	ap_memory	C_3_V	array

Σχήμα 26: Αποτελέσματα rtl

Cosimulation Report for 'OGmult_matr_func'

Result

RTL	Status	Latency			Interval		
		min	avg	max	min	avg	max
VHDL	Pass	4	4	4	NA	NA	NA
Verilog	NA	NA	NA	NA	NA	NA	NA

Export the report(.html) using the [Export Wizard](#)

Σχήμα 27: Αποτελέσματα rtl

Export Report for 'OGmult_matr_func

General Information

Report date: Sat Jun 05 21:49:21 +0300 2021
Project: VLSI2-EA5
Solution: solution3
Device target: xc7z030i-fbg484-2L
Implementation tool: Xilinx Vivado v.2019.2

Resource Usage

	VHDL
SLICE	0
LUT	486
FF	199
DSP	64
BRAM	0
SRL	0

Final Timing

	VHDL
CP required	10.000
CP achieved post-synthesis	3.975

Timing met

Export the report(.html) using the [Export Wizard](#)

Σχήμα 28: Αποτελέσματα rtl

Όπως παρατηρούμε μειώσαμε το latency σχεδόν 44.5 φορές αλλά με το κόστος να αυξήσουμε όχι απλά λίγο τους πόρους του συστήματός μας αλλά δραματικά αρκετά αφού όπως βλέπουμε τα DSPs αυξήθηκαν 64 φορές παραπάνω, τα FF να φτάνουν τα 1205 απο 30 και τα LUT να αγγίζουν τα 484 εναντι των 27. Στο τέλος ακολουθούν τα σχηματικά με την σειρά για τον κάθε κώδικα εκτός του τελευταίου λόγου των πραγματικά πολλών πόρων που το υλοποιούσανε.

